



TUGAS AKHIR - TE 141599

**Pengaturan Kecepatan Motor *Brushless* DC
dengan Menggunakan Metode PID STR**

Nitya Ayuning Fasalina
NRP 07111240000175

Dosen Pembimbing
Ir. Rusdhianto Effendie A.K., M.T.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

***Controlling Velocity of BrushlessDC Motor with
PID STR Method***

Nitya Ayuning Fasalina
NRP 07111240000175

Supervisor
Ir. Rusdhianto Effendie A.K., M.T.

***ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018***

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan tugas akhir saya dengan judul “**Pengaturan Kecepatan Motor Brushless DC dengan PID STR**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2018

Nitya Ayuning Fasalina
NRP 07111240000175

Halaman ini sengaja dikosongkan

**PENGATURAN KECEPATAN MOTOR BRUSHLESS DC
DENGAN PID STR**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik Elektro**

Pada

**Bidang Studi Teknik Sistem Pengaturan
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing,



Ir. Rusdhianto Effendie A.K., M.T.
NIP. 19570424 198502 1 001



Halaman ini sengaja dikosongkan

PENGATURAN KECEPATAN MOTOR BRUSHLESS DC DENGAN MENGGUNAKAN METODE PID STR

Nama : Nitya Ayuning Fasalina
Pembimbing : Ir. Rusdhianto Effendie A.K., M.T.

ABSTRAK

Motor DC, sebagai salah satu mesin listrik yang banyak digunakan di industri, penggunaannya banyak digantikan dengan motor *Brushless* DC (BLDC). Motor *Brushless* DC banyak dioperasikan di dunia industri, terutama industri yang mudah terbakar dan efisiensinya lebih tinggi daripada motor industri. Dalam berbagai penggunaan motor *Brushless* DC sering kali diperlukan kecepatan yang stabil seiring dengan beban yang berubah-ubah. Padahal untuk membuat kecepatan motor *Brushless* DC stabil tidaklah mudah, karena tidak adanya *brush* pada motor *Brushless* DC. Oleh karena itu diperlukan metode pengaturan kecepatan motor *Brushless* DC agar dapat mempertahankan kecepatan motor selama motor itu digunakan.

Pengendalian kecepatan pada motor *Brushless* DC bisa dilakukan dengan berbagai metode. Dalam Tugas Akhir ini metode yang digunakan adalah metode PID *Self Tuning Reguator*. Proportional Integral Derivative controller atau biasa disebut PID merupakan kontroler mekanisme umpan balik yang biasanya dipakai pada sistem kontrol industri. Dipilih kontroler PID karena kontrol yang paling sering dipakai. *Self Tuning Regulator* merupakan salah satu skema kontrol adaptif untuk kasus regulator.

Pada kondisi *plant* dengan tambahan beban sebanyak 50% hingga 100% pada beban *damper* dan beban inersia. Respon kecepatan memiliki nilai *Root Mean Squared Error (RMSE)* sebesar 0,0214. Dengan nilai RMSE yang relatif kecil atau mendekati 0, maka dapat disimpulkan bahwa perancangan Kontroler berhasil.

Kata Kunci: BLDC, Pengaturan Kecepatan Motor, PID, STR

Halaman ini sengaja dikosongkan

**CONTROLLING VELOCITY OF BRUSHLESS DC MOTOR
WITH PROPORTIONAL INTEGRAL DIFFERENTIAL (PID)
STR METHOD**

Name : Nitya Ayuning Fasalina
Supervisor : Ir. Rusdhianto Effendie A.K., M.T.

ABSTRACT

DC Motor, one of most used electrical machinery in industry, currently is replaced with Brushless DC Motor (BLDC). Brushless DC Motor mostly operated in extreme industrial area such as flammable environment. Besides, Brushless DC motor has higher level of efficiency than regular industrial motor. In practice, Brushless DC motor need to be operated in stable velocity, whatever the amount of the load is. However, to keep the velocity stable is not a simple thing because of the absence of brush. Thus, we need a method to control the velocity of the motor to keep it in a stable velocity during operation.

There are various method could be applied for this case, and in this final project, PID method will be applied. Proportional Integral Differential or abbreviated PID is a feedback mechanism controller which generally used in industrial control system. Thus, PID controller is chosen. In this final project, PID method is expected to control and to keep the stability of the Brushless DC motor velocity. Self Tuning Regulator itself is an adaptive scheme in the regulator case.

With an additional load of 50% to 100% on the dampening load and inertia load. The speed response has a Root Mean Squared Error (RMSE) value of 0.0214. With the RMSE value is relatively small or close to 0, it can be concluded that the controler designing is successful.

Keywords: *BLDC, Motor Speed Control, PID, STR*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Alhamdulillah, puji syukur penulis panjatkan kehadirat Allah SWT karena atas rahmat dan karunia-Nya penulis dapat menyelesaikan penulisan buku tugas akhir dengan judul **“Pengaturan Kecepatan Motor *Brushless* DC dengan Menggunakan Metode PID STR”**. Tugas akhir merupakan salah satu syarat yang harus dipenuhi untuk menyelesaikan program studi Strata-1 pada Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari bahwa dalam penulisan tugas akhir ini banyak mengalami kendala, namun berkat bantuan, bimbingan, dan kerja sama dari berbagai pihak sehingga kendala-kendala tersebut dapat di atasi. Untuk itu pada kesempatan ini penulis menyampaikan banyak terimakasih dan penghargaan setinggi-tingginya kepada :

1. Allah SWT. karena telah memberi hidayah serta inayah-Nya dalam menyelesaikan Tugas Akhir ini.
2. Kedua orang tua, Ayahanda Ibunda serta orang tua pengganti selama saya berkuliah di Surabaya yang telah memberikan banyak fasilitas yang mempermudah perkuliahan saya Ibu Sri Widayati dan Bapak Bambang Setiawan beserta keluarga yang selalu memberikan dukungan, semangat, dan doa kepada penulis.
3. Bapak Ir. Rusdhianto Effendie, M.T. selaku Dosen Pembimbing atas segala bantuan, perhatian, dan arahan selama pengerjaan tugas akhir ini.
4. Muhammad Yusron yang tak henti selalu memberikan doa, semangat, dan dukungannya.
5. Teman-teman seperjuangan Vira, Dwi, Tika, Fauziyah, Hanif, Pocay, Osa, Fahry. Serta Irin dan Novita.
6. Ayik, Tiffany, dan Renda yang telah banyak membantu di tahun terakhir hingga Tugas Akhir ini selesai.

7. Rekan-rekan e52 khususnya bidang studi Sistem Pengaturan yang banyak membantu penulis dalam memahami akademis dan teman teman lain yang banyak membantu penulis yang tidak dapat disebutkan satu persatu

Penulis berharap tugas akhir ini dapat bermanfaat bagi yang membutuhkannya.

Surabaya, Juni 2018

Penulis

DAFTAR ISI

PERNYATAAN KEASLIAN TUGAS AKHIR	v
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	3
1.5. Sistematika Penulisan.....	3
1.6. Relevansi.....	4
BAB 2 TINJAUAN PUSTAKA.....	5
2.1. Kajian Pustaka.....	5
2.2. <i>Brush Less DC Motor</i>	6
2.2.1. Jenis-jenis Motor <i>BLDC</i>	7
2.2.2. Model <i>BLDC Motor</i>	8
2.2.3. Konstanta Parameter	10
2.3. Kontroler.....	18
2.3.1. PID.....	18
2.3.2. Self Tuning Regulator (L1-STR)	19
2.4. MATLAB.....	21
BAB 3 PERANCANGAN SISTEM.....	25
3.1. Permodelan Motor BLDC	25
3.1.1. Permodelan Bagian Elektrik dan Mekanik Motor BLDC.....	25
3.1.2. Permodelan Beban pada Motor BLDC	27
3.1.3. Permodelan <i>Nominal Plant</i>	28
3.1.4. Diagram Blok <i>Simulink</i> Motor BLDC.....	28
3.2. Permodelan Kontroler.....	29
3.3. Desain Sistem Pengaturan.....	30
BAB 4.. SIMULASI SISTEM DAN ANALISA.....	31
4.1. Simulasi Motor BLDC	31

4.1.1. Kondisi <i>Open Loop</i> Tanpa Kontroler dengan Beban Nominal	31
4.1.2. Kondisi <i>Open Loop</i> Tanpa Kontroler dengan Beban Tambahan	32
4.1.3. Kondisi <i>Close Loop</i> dengan Kontroler dan Tanpa Beban Tambahan	33
4.1.4. Kondisi <i>Close Loop</i> dengan Kontroler dan Beban Tambahan	34
BAB 5 PENUTUP.....	37
5.1. Kesimpulan	37
5.2. Saran.....	37
DAFTAR PUSTAKA	39

DAFTAR GAMBAR

Gambar 2. 1 Model Ekuivalen BLDCM.....	10
Gambar 2. 2 Diagram Blok Sistem.....	22
Gambar 2. 3 Jendela <i>Front Panel</i> dari MATLAB	25
Gambar 3. 1 Diagram Blok <i>Simulink</i> Bagian Elektrik Motor.....	27
Gambar 3. 2 Diagram Blok <i>Simulink</i> Bagian Mekanik Motor.....	27
Gambar 3. 3 Diagram Blok <i>Simulink Load</i> pada Motor	28
Gambar 3. 4 Diagram Blok <i>Simulink Nominal Plant</i> pada Motor ..	29
Gambar 3. 5 Diagram Blok <i>Simulink</i> Motor BLDC	29
Gambar 3. 6 Desain Sistem Pengaturan.....	30
Gambar 4.1 Respon Kecepatan pada Kondisi <i>Open Loop</i> Tanpa Kontroler dan Tanpa Beban Tambahan.....	31
Gambar 4.2 Respon Kecepatan pada Kondisi <i>Open Loop</i> Tanpa Kontroler dan Beban Tambahan.....	32
Gambar 4.3 Respon Kecepatan pada Kondisi <i>Closed Loop</i> Dengan Kotroler dan Tidak Ada Beban Tambahan	33
Gambar 4.4 Respon <i>Output</i> antara <i>Plant</i> Model dan <i>Plant</i> Kondisi <i>Closed Loop</i> dengan Kontroler.....	34
Gambar 4.5 Respon Kecepatan pada Kondisi <i>Closed Loop</i> Dengan Kotroler dan Beban Tambahan.....	35
Gambar 4.6 Respon <i>Output</i> antara <i>Plant</i> Model dan <i>Plant</i> Kondisi <i>Closed Loop</i>	36
Gambar 4.6 Selisih Respon <i>Output</i> dari <i>Plant</i> Model Referensi	36

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 3. 1 Konstanta Parameter <i>Plant</i>	26
---	----

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

Pada Bab ini berisikan penjelasan awal mengenai penelitian yang akan dilakukan. Penjelasan tersebut meliputi latar belakang masalah, permasalahan yang diangkat, batasan masalah yang digunakan, tujuan dari penelitian ini, sistematika penulisan, penyusunan laporan dan juga relevansi.

1.1 Latar Belakang

Motor DC (*Dirrect Current*) merupakan salah satu jenis motor yang banyak diaplikasikan pada berbagai aplikasi industri. Misalnya dapat dijumpai sebagai penggerak beban mekanik. Motor DC yang digunakan pada bidang industri pada umumnya memiliki kapasitas daya yang relatif besar dan disesuaikan dengan beban mekanis serta volume produksi [1]. Namun secara bertahap penggunaan motor DC cenderung banyak digantikan oleh *Brush Less Dirrect Current Motor* (BLDCM), hal ini dikarenakan motor BLDC memiliki torsi awal yang tinggi dengan ukuran fisik yang lebih kecil, efisiensi tinggi, dan pemeliharaan lebih rendah karena tidak membutuhkan sikat komutator (*Brushless*) [2]. Dalam pengaplikasian motor BLDC, teknologi yang dikembangkan adalah dengan melakukan desain konstruksi motor, yaitu desain konstruksi mekanik, magnetik rotor, dan kumparan stator [3].

Namun selain memiliki berbagai kelebihan, motor BLDC memiliki pengendalian yang lebih rumit untuk pengontrolan kecepatan yang konstan karena tidak memiliki brush yang menunjang proses komutasi [4]. Pengontrolan kecepatan bertujuan untuk menghasilkan torsi yang konstan pula, sehingga *performance* dari motor tetap maksimal saat diberikan perubahan beban pada plant.

Beberapa penelitian terdahulu yang telah dilakukan, dirancang jenis konstruksi BLDCM medan aksial dual rotor single stator. Desain konstruksi kumparan stator dibuat seperti toroida, sehingga medan yang dihasilkan cenderung lebih banyak berinteraksi pada bagian dalam inti [5]. Kemudian Boldea, dkk melakukan penelitian mengenai desain konstruksi BLDCM dual rotor *single* stator dengan menerapkan dua metode pada desain kumparan stator, yaitu metode toroida dan radial. Selain itu desain konstruksi yang diterapkan yaitu menggabungkan metode *inner* rotor dengan metode *outer* rotor, sehingga medan yang dihasilkan oleh stator akan berinteraksi pada *inner* dan *outer*. Desain ini

dilengkapi dengan 12 buah magnet untuk setiap rotor, dan 18 kutub pada stator [6].

Pada penelitian [7], telah menerapkan desain konstruksi *modular multistage* pada rotor dan stator dan tida kelompok magnetik rotor (*third rotor dual stator*). Dimana pada setiap sisi rotor, terdiri dari sepuluh buah magnet permanen yang dipasang saling berlawanan kutub. Kemudian pada bagian stator, terdiri dari 12 kutub (satu fasa empat kutub) dengan desain kumparan stator yang dibuat searah.

Pada penelitian lainnya, dilakukan pengaturan kecepatan BLDCM menggunakan *hybrid sliding mode observer*. Pada risetnya, posisi rotor ditentukan berdasarkan hasil estimasi yang berasal dari pembacaan *hall* sensor dan arus stator. Dari hasil pembacaan tersebut, dapat diketahui kecepatan aktual rotor dan kemudian akan dibandingkan dari kecepatan referensi yang diberikan sebelumnya [8].

Berdasarkan permasalahan tersebut, maka diajukan skripsi mengenai topik energi terbarukan dengan judul "Pengaturan Kecepatan Motor *Brushless* DC dengan Menggunakan Metode *PID STR*", dimana pada penelitian ini dirancang desain konstruksi Pada penelitian ini juga dirancang kontroler yang berfungsi untuk mengontrol kecepatan putar motor BLDC yaitu menggunakan metode *Self Tuning Regulator* (STR) yang dapat mempertahankan kecepatan motor sesuai dengan setpoint yang telah ditentukan meskipun terjadi perubahan beban pada plant. Metode kontrol yang diterapkan dengan memperoleh parameter pada plant yang dianggap konstan, kemudian akan diselesaikan menyesuaikan parameter dari kontroler *Proportional Integral Derrivative* (PID) dengan tuning nilai K_p , K_i , dan K_d terhadap perubahan pembebanan yang diberikan pada plant.

1.2 Rumusan Masalah

Dari pemaparan pada subbab latar belakang, bahwa pada penelitian sebelumnya belum diterapkan desain kontroler pada BLDCM menggunakan metode adaptif *PID self tuning regulator* lebih spesifiknya *direct self tuning regulator*.

Oleh karena itu pada penelitian ini dilakukan perancangan desain kontroler pada BLDCM dengan menggunakan *plant* dan parameter dari penelitian sebelumnya. Kemudian pengontrolan kecepatan putaran BLDCM menggunakan metode *Adaptive PID direct STR* yang dapat menghasilkan kecepatan yang konstan sesuai dengan *setpoint* saat diberikan perubahan beban pada plant.

1.3 Batasan Masalah

Batasan masalah dari penelitian ini yaitu, pembahasan dalam ruang lingkup permodelan *plant Self Communication Multistage Axial Field BLDCM* dengan tujuan agar dapat dimodelkan seperti motor DC untuk kebutuhan simulasi, merancang desain kontroler *Adaptive PID direct STR*, menggabungkan desain kontrol kecepatan motor menggunakan metode *Adaptive PID direct STR* dengan *plant BLDC motor*. Melakukan berbagai percobaan simulasi dengan *MATLAB* terhadap *plant* dan kontroler dengan nilai beban *damp*er dan *inersia* yang berubah-ubah.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah merancang pengontrolan kecepatan menggunakan metode *Adaptive PID direct STR* yang dapat menghasilkan kecepatan yang konstan sesuai dengan setpoint saat diberikan perubahan beban pada *plant*.

1.5 Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam lima bab yang masing-masing bab akan membahas permasalahan yang berhubungan dengan Tugas Akhir yang telah dibuat dengan sistematika penulisan sebagai berikut:

1. BAB 1 PENDAHULUAN

Berisi tentang latar belakang, permasalahan, pembatasan masalah, tujuan, metodologi, sistematika penulisan dan relevansi terhadap pembahasan Tugas Akhir ini, yaitu, control kecepatan motor *BLDC* dengan menggunakan *adaptive kontroler PID direct STR*.

2. BAB 2 TINJAUAN PUSTAKA

Menjelaskan tentang teori-teori dasar yang melandasi pembuatan Tugas Akhir ini yang meliputi: gambaran umum mengenai pemodelan sistem, desain konstruksi *BLDCM* medan aksial, dan kontroler *adaptive PID direct Self Tunning Regulator (STR)* untuk pengontrolan kecepatan pada *plant* motor *BLDC*.

3. BAB 3 PERANCANGAN SISTEM

Perancangan sistem yang dilakukan meliputi permodelan matematika dari *BLDCM* menjadi motor DC biasa untuk kebutuhan simulasi sistem. Setelah mendapat berbagai

konstanta parameter-parameter dari *plant* dibuat permodelan kontroler yang akan dibuat menggunakan *PID Self Tunning Regulator* (STR).

4. **BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM**

Pada tahap ini dilakukan percobaan terhadap *plant* (BLDCM) baik yang belum dilengkapi dengan kontroler maupun yang telah dilengkapi dengan kontroler *adaptive PID direct Self Tunning Regulator* (STR). Kemudian mengamati bagaimana performa dari motor jika diberikan pembebanan tambahan yang berubah-ubah pada bagian beban *dampner* dan inersia.

5. **BAB 5 PENUTUP**

Kesimpulan diperoleh sesuai dengan hasil pengujian dan analisis yang dilakukan. Dari hasil penelitian yang telah dilakukan akan diperoleh kekurangan dari penelitian yang sudah dilakukan, sehingga didapatkan saran untuk pengembangan penelitian selanjutnya.

1.6 Relevansi

Kontribusi penelitian yang ingin dicapai adalah untuk mendapatkan desain *plant Self Communication Multistage Axial Field* BLDCM yang digunakan untuk simulasi karena BLDCM dapat dimodelkan seperti motor DC biasa. Serta dilengkapi dengan pengontrolan kecepatan putaran motor menggunakan metode *PID Self Tunning Regulator* (STR) yang dapat menghasilkan kecepatan yang konstan sesuai dengan setpoint saat diberikan perubahan beban tambahan pada *plant*. Dari beberapa hal tersebut, diharapkan menjadi kontribusi dari terhadap perkembangan metode kontrol terhadap *Self Communication Multistage Axial Field* BLDCM dengan metode *PID direct Self Tunning Regulator* (STR).

BAB 2

TINJAUAN PUSTAKA

Pada Bab ini, berisi tentang teori penunjang dari berbagai pustaka atau literatur seperti buku dan *paper* yang mendukung penelitian ini. Teori penunjang pada Bab ini meliputi gambaran umum mengenai Sistem kontrol yang dibuat meliputi pembuatan permodelan *BLDCM* dengan tujuan agar dapat dimodelkan seperti motor DC untuk kebutuhan simulasi pada *SIMULINK*, setelah itu dilakukan pembuatan desain kontroler menggunakan metode kontrol *Adaptive PID Dirrect Self Tunning Regulator* (STR) yang akan diterapkan pada *plant Self Communication Multistage Axial Field BLDCM*. Dimana sistem kontrol yang dibuat bertujuan untuk mempertahankan kecepatan putaran motor sesuai dengan *set point* yang ditentukan ketika terdapat perubahan beban berupa *dampner* (*B*) dan *momen inersia* (*J*).

2.1 Kajian Pustaka

Pembuatan permodelan *plant* BLDCM dibuat untuk kebutuhan simulasi menggunakan *SIMULINK* dan desain kontroller dibuat bertujuan untuk mengontrolan kecepatan putaran BLDCM saat terdapat perubahan beban berupa *dampner* dan *momen inersia* (*J*). Namun terdapat perbedaan dengan penelitian terdahulu yang terletak pada metode yang digunakan. Adapun permodelan *plant* dan sistem kontrol yang pernah dibuat akan dijelaskan pada subbab selanjutnya.

Perancangan jenis konstruksi motor BLDC medan aksial *dual rotor single stator*. Desain konstruksi kumparan *stator* dibuat seperti *toroida*, sehingga medan yang dihasilkan cenderung lebih banyak berinteraksi pada bagian dalam inti [5]. Kemudian Boldea,dkk melakukan penelitian mengenai desain konstruksi BLDCM *dual rotor single stator* dengan menerapkan dua metode pada desain kumparan *stator*, yaitu metode *toroida* dan *radial*. Selain itu desain konstruksi yang diterapkan yaitu menggabungkan metode *inner rotor* dengan metode *outer rotor*, sehingga medan yang dihasilkan oleh *stator* akan berinteraksi pada *inner* dan *outer*. Desain ini dilengkapi dengan 12 buah magnet untuk setiap *rotor*, dan 18 kutub pada *stator* [6].

Pada penelitian [7], telah menerapkan desain konstruksi *modular multistage* pada *rotor* dan *stator* dan tida kelompok magnetik *rotor* (*third rotor dual stator*). Dimana pada setiap sisi *rotor*, terdiri dari sepuluh buah

magnet permanen yang dipasang saling berlawanan kutub. Kemudian pada bagian *stator*, terdiri dari 12 kutub (satu fasa empat kutub) dengan desain kumparan *stator* yang dibuat searah.

Pada penelitian lainnya, dilakukan pengaturan kecepatan BLDCM menggunakan *hybrid sliding mode observer*. Pada risetnya, posisi *rotor* ditentukan berdasarkan hasil estimasi yang berasal dari pembacaan *hall sensor* dan arus *stator*. Dari hasil pembacaan tersebut, dapat diketahui kecepatan aktual *rotor* dan kemudian akan dibandingkan dari kecepatan referensi yang diberikan sebelumnya [8].

2.2 Brush Less DC Motor

Motor *Brushless DC* merupakan salah satu jenis motor sinkron magnet permanen yang disuplai oleh sumber listrik DC pada bagian kontrolnya, selain itu membutuhkan sumber listrik AC tiga fasa untuk menggerakkan bagian rotor. Sumber listrik AC tiga fasa dibutuhkan karena motor sinkron magnet permanen ini memiliki 3 buah koil pada stator, kemudian hubungan antar koil dan belitan stator trapezoidal akan memberikan *elektromotif back trapezoidal* (gaya gerak listrik balik trapezoidal) yaitu tegangan balik yang dihasilkan oleh belitan motor *brushless DC* yang akan menggerakkan rotor. Pergerakan pada rotor ini disebabkan oleh medan magnet pada stator yang menerima suplai adalah bagian dua fasanya, sedangkan satu fasa lainnya tidak tersuplai. Maka dari itu fenomena ini mengakibatkan motor ini seperti motor DC, karena arus yang mengalir pada kumparan stator mirip dengan motor DC meskipun sebenarnya motor yang digunakan adalah berjenis tiga fasa.

Motor *BLDC* ini menggunakan sistem komutasi elektrik atau sering disebut dengan *electronically commutated motor*. Sistem komutasi elektrik ini diartikan sebagai fungsi dari *switch electronic*. Komutator elektronik ini terdiri dari kombinasi transistor atau biasanya dapat juga menggunakan MOSFET atau IGBT yang membutuhkan sinyal atau pulsa dan dalam hal pengaktifan koil dapat dilakukan dalam waktu yang tepat sehingga hal ini dapat menggerakkan motor.

Namun jika dibandingkan dengan motor DC biasa, motor *BLDC* memiliki kelebihan lainnya, yaitu memiliki efisiensi tinggi, konsumsi energi lebih rendah, umur penggunaan yang relatif lebih panjang, dan tidak menimbulkan *electrical noise*. Secara umum konstruksi motor *BLDC* terdiri dari *stator* dan *rotor*. Konstruksi dasar dari sebuah stator yaitu memiliki kumparan tiga fasa, jumlah kumparan dibuat untuk memiliki riak torsi yang lebih kecil. Kemudian pada setiap kumparan

stator masing-masing memiliki tiga unsur yang terhubung seri, yaitu induktansi (L), resistansi (R), dan gaya gerak listrik kembali (*back e.m.f*) [3].

Oleh karena tidak adanya brush pada motor BLDC, untuk menentukan timing komutasi yang tepat pada motor ini sehingga didapatkan torsi dan kecepatan yang konstan, diperlukan 3 buah sensor *Hall* dan atau *encoder*. Pada sensor *Hall*, *timing* komutasi ditentukan dengan cara mendeteksi medan magnet rotor dengan menggunakan 3 buah sensor Hall untuk mendapatkan 6 kombinasi *timing* yang berbeda, sedangkan pada *encoder*, *timing* komutasi ditentukan dengan cara menghitung jumlah pola yang ada pada encoder.

Pada umumnya *encoder* lebih banyak digunakan pada motor BLDC komersial karena *encoder* cenderung mampu menentukan *timing* komutasi lebih presisi dibandingkan dengan menggunakan sensor hall. Hal ini terjadi karena pada *encoder*, kode komutasi telah ditetapkan secara *fixed* berdasarkan banyak *pole* dari motor dan kode inilah yang digunakan untuk menentukan timing komutasi. Namun karena kode komutasi *encoder* untuk suatu motor tidak dapat digunakan untuk motor dengan jumlah *pole* yang berbeda. Hal ini berbeda dengan sensor *Hall*. Apabila terjadi perubahan *pole* rotor pada motor, posisi sensor *hall* dapat diubah dengan mudah. Hanya saja kelemahan dari sensor *hall* adalah apabila posisi sensor *hall* tidak tepat akan terjadi kesalahan dalam penentuan timing komutasi atau bahkan tidak didapatkan 6 kombinasi *timing* komutasi yang berbeda [9].

2.2.1 Jenis-jenis Motor BLDC [10]

a. Jenis Inner Rotor

Motor jenis ini merupakan motor jenis normal, dimana bagian rotor dikelilingi oleh inti dan kumparan stator. Berdasarkan penamaannya, menunjukkan bahwa yang berputar sebagai rotor adalah pada bagian dalam motor (*inner rotor*).

b. Jenis Outer Rotor

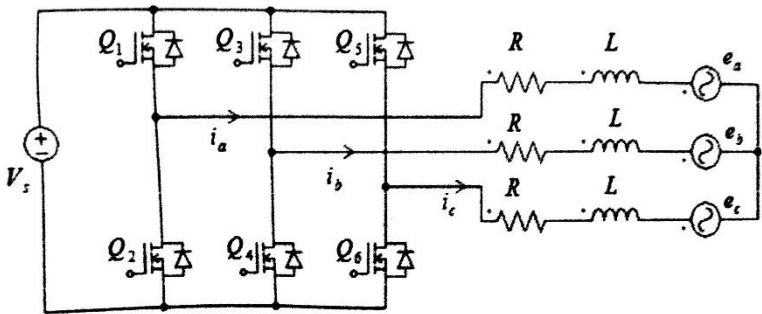
Motor jenis ini merupakan jenis motor dimana bagian inti dan kumparan stator dikelilingi oleh rotor, atau biasa disebut dengan jenis terbalik (*invers*). Berdasarkan jenis penamaannya menunjukkan bahwa yang berputar sebagai rotor adalah pada bagian luar motor (*outer rotor*). Berikut adalah bentuk konstruksi motor jenis *outer rotor*.

c. **Jenis Axial Gap**

Motor jenis ini merupakan jenis datar, dimana magnet permanen rotor ditempatkan saling berhadapan dengan kumparan stator dalam suatu bidang datar. Jenis motor ini paling banyak didesain tanpa inti (*coreless*) pada bagian kumparan *stator*, hal ini bertujuan untuk menghilangkan *cogging*.

2.2.2 Model BLDC Motor [3]

Permodelan yang dilakukan pada konstruksi *BLDC Motor* dilakukan untuk memudahkan perhitungan yang dipresentasikan dalam model matematika. Berdasarkan Gambar 2.1, *BLDC Motor* dapat dipresentasikan dalam bentuk persamaan.



Gambar 2. 1 Model ekuivalen BLDCM

Persamaan yang diperoleh dari gambar 2.1 didapat persamaan tiga fasa. Persamaan tegangan dari *BLDC Motor* dapat dinyatakan dengan Persamaan (2.1).

$$\begin{aligned}
 (L + 2M) \frac{d(i_U + i_V + i_W)}{dt} + R(i_U + i_V + i_W) + (e_U + e_V + e_W) \\
 = (V_U + V_V + V_W) - 3v_N
 \end{aligned} \tag{2.1}$$

dengan:

i_U, i_V, i_W : arus pada setiap fasa

e_U, e_V, e_W : *back e.m.f* pada setiap fasa
 L : induktansi bersama antara kedua fasa
 M : induktansi bersama antara kedua fasa
 R : resistansi pada setiap fasa
 V_U, V_V, V_W : tegangan pada setiap fasa
 v_N : tegangan pada titik netral

Dengan mempertimbangkan pengaruh *back e.m.f* pada masing masing fasa, sehingga dapat diuraikan persamaan berikut.

$$L \frac{di_U}{dt} + Ri_U + M \left(\frac{di_V}{dt} + \frac{di_W}{dt} \right) + \omega K_{AC} \cos(p\theta + \delta) = V_U - v_N \quad (2.2a)$$

$$L \frac{di_V}{dt} + Ri_V + M \left(\frac{di_W}{dt} + \frac{di_U}{dt} \right) + \omega K_{AC} \cos(p\theta + \delta - 2\pi/3) = V_V - v_N \quad (2.2b)$$

$$L \frac{di_W}{dt} + Ri_W + M \left(\frac{di_U}{dt} + \frac{di_V}{dt} \right) + \omega K_{AC} \cos(p\theta + \delta - 2\pi/3) = V_W - v_N \quad (2.2c)$$

dengan:

K_{AC} : tegangan *back e.m.f* konstan pada setiap fasa
 p : jumlah kutub
 θ : posisi sudut rotor
 δ : sudut fasa
 ω : kecepatan rotor

Oleh karena itu, untuk *back e.m.f* pada setiap fasa didefinisikan

$$e_U = \omega K_{AC} \cos(p\theta + \delta) \quad (2.3a)$$

$$e_V = \omega K_{AC} \cos(p\theta + \delta - 2\pi/3) \quad (2.3b)$$

$$e_W = \omega K_{AC} \cos(p\theta + \delta - 2\pi/3) \quad (2.3c)$$

Dengan mensubstitusikan Persamaan (2.2abc) dan Persamaan (2.3abc) maka didapatkan Persamaan (2.4abc).

$$L \frac{di_U}{dt} + Ri_U + M \left(\frac{di_V}{dt} + \frac{di_W}{dt} \right) + e_U = \quad (2.4a)$$

$$L \frac{di_U}{dt} + Ri_U + M \left(\frac{di_V}{dt} + \frac{di_W}{dt} \right) + e_V = \quad (2.4b)$$

$$L \frac{di_w}{dt} + Ri_w + M \left(\frac{di_u}{dt} + \frac{di_v}{dt} \right) + e_w \quad (2.4c)$$

Dari Persamaan (2.4abc), maka dapat dibuat dalam bentuk matriks seperti yang ditunjukkan pada Persamaan (2.5)

$$\begin{bmatrix} L & M & L \\ M & L & M \\ M & M & L \end{bmatrix} \begin{bmatrix} \frac{di_u}{dt} \\ \frac{di_v}{dt} \\ \frac{di_w}{dt} \end{bmatrix} = \begin{bmatrix} -R & 0 & 0 \\ 0 & -R & 0 \\ 0 & 0 & -R \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_u \\ V_v \\ V_w \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_u \\ e_v \\ e_w \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} v_n \quad (2.5)$$

2.2.3 Konstanta Parameter[3]

Seluruh konstanta parameter yang diperlukan dari plant didapatkan dari referensi [3]. Adapun konstanta parameter yang diperlukan yaitu resistansi (R), induktansi (L), konstanta torsi (K_{Tm}), konstanta ggl *back e.m.f* (K_{ggl}), induktansi bersama atau *mutual inductance* (M), *dampner* (B), momen inersia (J).

2.2.3.1 Resistansi dan Induktansi (L)

Kumparan stator atau kumparan medan, adalah salah satu bagian utama pada setiap motor. Bagian ini terdiri dari gulungan kawat tembaga yang mempunyai resistansi (R) dan induktansi (L). Pada proses ini dilakukan pengukuran secara langsung pada kumparan stator, dengan tujuan untuk mendapatkan nilai konstanta parameter R dan L. adapun proses pengukuran yaitu dengan menggunakan LCR meter.

2.2.3.2 Konstanta Torsi (K_{Tm})

Identifikasi untuk mendapatkan nilai K_{Tm} dari *plant*, yaitu dengan melalui beberapa tahap

$$T_m = K_{Tm} (i_a + i_b + i_c) \quad (2.6)$$

Untuk kondisi tanpa beban (beban nol)

$$P_{in_0} = V_{in_0} \cdot i_{in_0} \quad (2.7)$$

Pada Persamaan (2.8), P_{r_0} didefinisikan sebagai rugi rugi pada plant dengan kondisi tanpa beban, yaitu rugi rugi elekttrtik ($i_0^2 R$) dan rugi rugi mekanik (P_{rm})

$$P_{in_0} = P_{O_0} + P_{r_0} \rightarrow (P_{r_0} = i_0^2 R + P_{rm}) \quad (2.8)$$

dapat didefinisikan menjadi Persamaan (2.9)

$$P_{in_0} = P_{O_0} + i_0^2 R + P_{rm} \quad (2.9)$$

Karena pada plant terdiri dari empat kelompok kumparan stator untuk setiap fasa, maka berlaku persamaan.

$$P_{in_0} = (V_{in_0} \cdot i_{in_0}) + 12i_0^2 R + P_{rm} \quad (2.10)$$

atau

$$P_{in_0} = \omega_0 T m_0 + 12i_0^2 R + P_{rm} \quad (2.11)$$

untuk kondisi berbeban

$$P_{in_*} = V_{in_*} \cdot i_{in_*} \quad (2.12)$$

Pada Persamaan (2.13), P_{r_*} didefinisikan sebagai rugi-rugi pada *plant* dengan kondisi berbeban, yaitu rugi-rugi elektrik ($i_*^2 R$) dan rugi-rugi mekanik (P_{rm}).

$$P_{in_*} = P_{O_*} + P_{r_*} \rightarrow (P_{r_*} = i_*^2 R + P_{rm}) \quad (2.13)$$

Dapat difenisikan menjadi Persamaan (2.14)

$$P_{in_*} = P_{O_*} + i_*^2 R + P_{rm} \quad (2.14)$$

Karena pada *plant* terdiri dari empat kelompok kumparan stator untuk setiap fasa, maka berlaku Persamaan (3.15)

$$P_{in_*} = (V_{in_*} \cdot i_{in_*}) + 12i_*^2 R + P_{rm} \quad (2.15)$$

atau

$$P_{in_*} = \omega_0 Tm_* + 12i_*^2 R + P_{rm} \quad (2.16)$$

dengan:

Tm	: torsi motor
Tm_0	: torsi motor pada kondisi tanpa beban
Tm_*	: torsi motor pada kondisi berbeban
K_{Tm}	: konstanta torsi motor
i_a, i_b, i_c	: arus pada setiap fasa
P_{in_0}	: daya input pada kondisi berbeban
P_{in_*}	: daya input pada kondisi berbeban
V_{in_0}	: tegangan input pada kondisi tanpa beban
V_{in_*}	: tegangan input pada kondisi berbeban
i_{in_0}	: arus input pada kondisi tanpa beban
i_{in_*}	: arus input pada kondisi berbeban
P_{O_0}	: daya output pada kondisi tanpa beban
P_{O_*}	: daya output pada kondisi berbeban
P_{r_0}	: rugi-rugi daya pada kondisi tanpa beban
P_{r_*}	: rugi-rugi daya pada kondisi berbeban
P_{rm}	: rugi-rugi daya mekanik
ω_0	: output kecepatan actual

Berdasarkan dua kondisi,yaitu tanpa beban dan berbeban maka dapat didefinisikan ΔTm seperti Persamaan (2.17)

$$\Delta Tm = \frac{1}{\omega_0} (V_{in_*} \cdot i_{in_*} - V_{in_0} \cdot i_{in_0}) - 12(i_*^2 - i_0^2)R \quad (2.17)$$

atau

$$\Delta Tm = \frac{1}{\omega_0} \Delta P_{in} - 12(i_*^2 - i_0^2)R \quad (2.18)$$

atau

$$\Delta Tm = K_{Tm} - (\Delta i)R \quad (2.19)$$

K_{Tm} bisa didapatkan dengan Persamaan (2.20)

$$K_{Tm} = \frac{\Delta Tm}{(\Delta i)R} = \frac{\Delta Tm}{(i_* - i_o)R} \quad (2.20)$$

2.2.3.3 Konstanta ggl / *Back e.m.f* (K_{ggl})

Pada proses identifikasi untuk mendapatkan konstanta ggl (K_{ggl}), BLDCM akan difungsikan sebagai generator. Proses ini menggunakan motor induksi sebagai penggerak utama (*prime mover*), dengan variasi kecepatan antara 300 rpm s/d 750 rpm pada range kecepatan setiap 50 rpm. Perubahan kecepatan 50 rpm, akan dicatat hasil pembacaan tegangan output (V_{ggl}) yang ditampilkan pada *Voltmeter*. Adapun alur proses untuk mendapatkan nilai K_{ggl} , yaitu dengan menurunkan persamaan sebagai berikut.

$$e = B \cdot L \cdot V \cdot \sin \alpha \quad (2.21)$$

Karena $\alpha = 90^\circ$; $\sin 90^\circ = 1$, maka

$$e = B \cdot L \cdot V \quad (2.22)$$

Untuk satu kumparan (2 kawat), maka

$$e = 2 \cdot B \cdot L \cdot V \quad (2.23)$$

Untuk n kumparan, maka

$$e = 2 \cdot n \cdot B \cdot L \cdot V \quad (2.24)$$

Kumparan yang memotong garis gaya medan magnet yang berputar dengan jari-jari r, pada kecepatan ω , maka pada kedua ujung kumparan tersebut akan menimbulkan tegangan (V). Didefinisikan dengan persamaan sebagai berikut.

$$V = \omega \cdot r \quad (2.25)$$

Fluks yang dibangkitkan

$$\varphi = \frac{B}{A} \quad (2.26)$$

maka

$$B = A \cdot \varphi \quad (2.27)$$

Dapat didefinisikan

$$e = 2 \cdot n_l \cdot (A \cdot \varphi) \cdot L \cdot (\omega r) \quad (2.28)$$

atau

$$e = \frac{2 \cdot n_l \cdot A \cdot L \cdot r}{\omega} \cdot n \cdot \varphi \rightarrow \frac{2 \cdot n_l \cdot A \cdot L \cdot r}{\omega} = C \quad (2.29)$$

V_{ggl} dapat didefinisikan

$$e = C \cdot n \varphi \quad (2.30)$$

atau

$$e = K_{ggl} \cdot \omega \quad (2.31)$$

K_{ggl} dapat didefinisikan

$$K_{ggl} = \frac{e}{\omega} \quad (2.32)$$

2.2.3.4 Induktansi bersama (M)

Identifikasi akan dilakukan untuk mendapatkan besaran induktansi bersama atau *mutual inductance* (M) antara kumparan-kumparan stator pada *plant*.

Pada sisi sekunder, berlaku persamaan tegangan terhadap arus seperti ditunjukkan persamaan berikut

$$V_0 = -L \frac{di_2}{dt} + M \frac{di_1}{dt} \quad (2.33)$$

karena sisi sekunder *open loop*, maka $i_2 = 0$ sehingga Persamaan (2.33) dapat ditulis menjadi Persamaan (2.34)

$$V_0 = M \frac{di_1}{dt} \quad (2.34)$$

Pada sisi primer berlaku Persamaan (2.35)

$$V_{in} - R_i i_1 - L \frac{di_1}{dt} - M \frac{di_2}{dt} = 0 \quad (2.35)$$

Oleh karena sisi sekunder *open loop*, maka $i_2 = 0$, sehingga persamaan (2.35) dapat ditulis menjadi Persamaan (2.36)

$$V_{in} - R_i i_1 - L \frac{di_1}{dt} = 0 \quad (2.36)$$

atau

$$L \frac{di_1}{dt} = V_{in} - R_i i_1 = 0 \quad (2.37)$$

atau

$$\frac{di_1}{dt} = \frac{1}{L} (V_{in} - R_i i_1) = 0 \quad (2.38)$$

Substitusi Persamaan (2.33) dan (2.37) ke Persamaan (2.39).

$$V_0 = M \left\{ \frac{1}{L} (V_{in} - R_i i_1) \right\} \quad (2.39)$$

Untuk mendapatkan konstanta (M) digunakan Persamaan (2.40)

$$M = \frac{L \cdot V_0}{V_{in} - R_i i_1} \quad (2.40)$$

2.2.3.5 Damper (B)

Menentukan konstanta *damper* (B). Pada poros motor yang berputar pada kecepatan ω , dari gaya dibangkitkan T_m , momen inersia J, dari gaya lawan oleh torsi beban T_L dapat dituliskan dalam persamaan sebagai berikut

$$T_m = J \frac{d\omega}{dt} + B\omega + T_L \quad (2.41)$$

Karena pada saat identifikasi kondisi motor tanpa beban, sehingga $T_L = 0$. Maka persamaan dapat ditulis menjadi

$$T_m = J \frac{d\omega}{dt} + B\omega \quad (2.42)$$

atau

$$T_m = K_{Tm} \times i_T \quad (2.43)$$

Pada keadaan *steady state* $\frac{d\omega}{dt} = 0$, maka persamaan dapat ditulis menjadi

$$K_{Tm} \times i_T = J \cdot 0 + B\omega \quad (2.44)$$

Berdasarkan persamaan (2.44), maka konstanta *damper* dapat didefinisikan sebagai berikut

$$B = \frac{K_{Tm} \times i_T}{\omega} \quad (2.45)$$

2.2.3.6 Moment Inersia (J)

Menentukan nilai konstanta J untuk $\omega(k)$ dengan nilai kecepatan 0 s/d 460 rpm. Untuk persamaan torsi motor (T_m) pada kondisi tanpa beban, maka dapat digunakan Persamaan (2.42)

$$T_m = J \frac{d\omega}{dt} + B\omega$$

Persamaan Torsi T_m terhadap inersia dan *damper* ditunjukkan pada persamaan sebagai berikut

$$T_m(k) = J \frac{\omega(k) - \omega(k-1)}{\Delta t} + B\omega(k) \quad (2.46)$$

atau

$$T_m(k) = \frac{J}{\Delta t} \omega(k) - \frac{J}{\Delta t} \omega(k-1) + B\omega(k) \quad (2.47)$$

Mendefinisikan α dan β dari Persamaan (2.48)

$$T_m(k) = \underbrace{\left(\frac{J}{\Delta t} + B\right)}_{\alpha} \omega(k) - \underbrace{\left(\frac{J}{\Delta t}\right)}_{\beta} \omega(k-1) \quad (2.48)$$

Dilakukan percobaan dengan mengapati respon transient kecepatan dan arus pada real plant dengan Δt 40 ms. Sehingga didapatkan 243 data. berdasarkan hasil respon tersebut, sehingga dapat dilakukan identifikasi untuk mengetahui nilai konstanta momen insersia (J) dengan memasukan data respon pada setiap perubahan 40 ms.

$$\alpha \omega(k) - \beta \omega(k-1) = T_m(k) \quad (2.49)$$

$$[\omega(k) - \omega(k-1)] \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = T_m(k) \quad (2.50)$$

Persamaan (2.49) dan (2.50) dapat dituliskan sebagai berikut

$$\overbrace{\begin{bmatrix} \omega(2) & -\omega(1) \\ \vdots & \vdots \\ \omega(243) & -\omega(242) \end{bmatrix}}^Q \underbrace{\begin{bmatrix} \alpha \\ \beta \end{bmatrix}}_x = \overbrace{\begin{bmatrix} T_m(2) \\ \vdots \\ T_m(243) \end{bmatrix}}^P \quad (2.51)$$

Maka nilai α dan β dapat dicari dengan persamaan

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = (Q^T Q)^{-1} Q^T P \quad (2.52)$$

Dari Persamaan (2.52) didapatkan nilai β , sehingga didapatkan Persamaan (2.53) untuk mencari nilai J.

$$J = \beta \cdot \Delta t \quad (2.53)$$

2.3 Kontroler

Pada pengerjaan tugas akhir ini digunakan kontroler adaptive PID direct STR. Penjelasan yang lebih lengkap akan dibahas pada subab selanjutnya.

2.3.1 PID [14]

PID adalah singkatan dari Proportional-Integral-Derivative kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu *proportional*, *integral*, dan *derivative*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang diinginkan terhadap suatu plant.

2.3.1.1 Kontrol P

Kp adalah Konstanta Proporsional. Kp berlaku Sebagai Gain atau penguat saja tanpa memberikan efek dinamik kepada kinerja kontroler. Penggunaan kontrol P memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini. Walaupun demikian dalam aplikasi-aplikasi dasar yang sederhana kontrol P ini cukup mampu untuk memperbaiki respon transien khususnya *rise time* dan *settling time*.

Pengontrol proporsional memiliki keluaran yang sebanding sebanding proporsional dengan besarnya sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya).

2.3.1.2 Kontrol I

Pengontrol Integral berfungsi menghasilkan respon sistem yang memiliki kesalahan *steady state* nol atau nilai *error steady state* adalah nol. Jika sebuah pengontrol tidak memiliki unsur integrator, pengontrol proporsional tidak mampu menjamin keluaran sistem dengan *error steady state* nol.

Ki adalah konstanta Integral, akan menjadi sangat besar sehinggadiharapkan dapat memperbaiki *error*. Jika $e(T)$ mendekati nol maka efek kontrol I ini semakin kecil.

Kontrol I dapat memperbaiki sekaligus menghilangkan respon *steady-state*, namun pemilihan Ki yang tidak tepat dapat menyebabkan respon transien yang tinggi sehingga dapat menyebabkan ketidak stabilan

sistem. Pemilihan K_i yang sangat tinggi justru dapat menyebabkan *output* berosilasi karena menambah orde sistem keluaran. pengontrol ini merupakan hasil penjumlahan yang terus menerus dari perubahan *input*.

Jika sinyal kesalahan tidak mengalami perubahan, maka keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Sinyal keluaran pengontrol integral merupakan luas bidang yang dibentuk oleh kurva kesalahan atau *error*.

2.3.1.3 Kontrol D

Keluaran pengontrol diferensial memiliki sifat seperti halnya suatu operasi derivatif. Perubahan yang mendadak pada masukan pengontrol akan mengakibatkan perubahan yang sangat besar dan cepat. Ketika masukannya tidak mengalami perubahan, keluaran pengontrol juga tidak mengalami perubahan, sedangkan apabila sinyal masukan berubah mendadak dan menaik (berbentuk fungsi *step*), keluaran menghasilkan sinyal berbentuk impuls. Jika sinyal masukan berubah naik secara perlahan (fungsi *ramp*), keluarannya justru merupakan fungsi *step* yang besar magnitudenya sangat dipengaruhi oleh kecepatan naik dari fungsi *ramp* dan factor konstanta K_d .

$$u(t) = K_p \tau_d \frac{de(t)}{dt} \quad (2.54)$$

Sinyal kontrol u yang dihasilkan oleh kontrol D dapat dinyatakan dari persamaan di atas, nampak bahwa sifat dari kontrol D ini dalam konteks “kecepatan” atau rate dari *error*. Dengan sifat ini ia dapat digunakan untuk memperbaiki respon transien dengan memprediksi error yang akan terjadi. Kontrol *Derivative* hanya berubah saat ada perubahan *error*, sehingga saat *error* statis kontrol ini tidak akan bereaksi, hal ini pula yang menyebabkan kontroler *Derivative* tidak dapat dipakai sendiri.

2.3.2 Self tuning regulator LI (LI – STR)

STR merupakan salah satu skema kontrol adaptif untuk kasus regulator. Secara umum suatu sistem STR memiliki mekanisme adaptasi dimana parameter kontroler selaras dengan perubahan parameter plant. Atau dengan kata lain aksi kontrol suatu sistem STR memiliki mekanisme

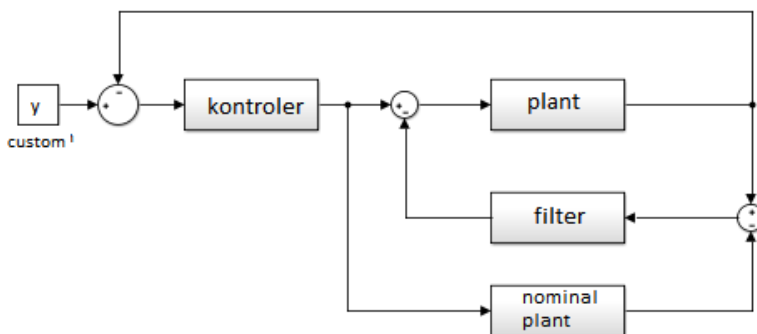
untuk melakukan proses kompensasi, yang berguna untuk mengatasi perubahan parameter maupun gangguan yang ditanggung plant.

Berdasarkan pada mekanisme penyesuaian parameter kontroler maupun maupun proses perhitungan sinyal kontrol, STR dibedakan atas 2 tipe yaitu *direct STR* dan *indirect STR*.

Pada suatu sistem *STR indirect*, parameter kontroler diperhitungkan berdasarkan nilai parameter *plant* beserta spesifikasi performansi yang ingin dicapai. Dimana nilai parameter *plant* didapat melalui proses identifikasi parameter.

Sedangkan pada sistem *STR direct*, parameter kontroler diestimasi berdasarkan spesifikasi eror model yang akan dipenuhi. Di samping itu terdapat pula mekanisme lain, yaitu suatu mekanisme yang menghasilkan sinyal kontrol kompensasi yang memaksa sistem closed loop selalu memenuhi performansi yang diinginkan.

Kontrol adaptif *L1-STR*, merupakan kontrol *Adaptive STR-Direct* dimana sinyal kompensasi yang dihasilkan dibentuk berdasarkan perbedaan sinyal *error* sementara nominal dengan *plant*. Kontrol *Adaptive L1-STR* dapat digambarkan sebagai berikut



Gambar 2. 2 Kontrol Adaptif L1-STR

Nominal *plant* adalah model dinamik *plant* sistem dengan parameter nominal. Dapat berupa model dinamik linier atau non linier sesuai dengan model dinamik *plant*.

Kontroler dapat berupa kontroler *PID* yang di desain dengan metode tertentu sedemikian rupa sehingga respons sistem closed loop memenuhi spesifikasi performansi yang diinginkan.

Filter *error feedback* menggunakan filter *LPF*, dimana orde filter boleh sama dengan orde plant atau satu derajat di bawahnya. Model dinamik *filter LPF* dapat ditulis sebagai berikut.

$$G(s) = \frac{1}{P(z)} \Bigg|_{z=\frac{s}{\omega_c}} \quad (2.55)$$

Dimana

$P(z)$: Polinomial butterword
 $\omega_c : 2\pi f_c$, f_c = frekuensi *cut-off*

Filter *LPF butterword* orde 1 dan orde 2 dapat dituliskan sebagai berikut

$$G_1(s) = \frac{1}{z + 1} \Bigg|_{z=\frac{s}{\omega_c}} \quad (2.56)$$

$$G_1(s) = \frac{1}{z^2 + \sqrt{2}z + 1} \Bigg|_{z=\frac{s}{\omega_c}} \quad (2.57)$$

Dimana

$\omega_c : 2\pi f_c$, f_c = fr *Self tuning regulator L1 (L1 - STR* fekuensi *cut-off*).

Frekusensi *cut-off filter* dipilih sekurang kurangnya tiga kali nilai pole dominan sistem hasil desain atau tiga kali *time costant* respon sistem hasil desain.

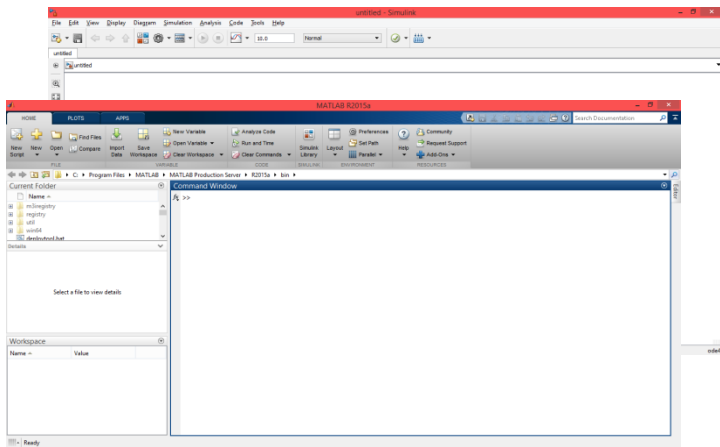
2.4 MATLAB [13]

MATLAB merupakan paket program dengan bahasa pemrograman yang tinggi untuk mengembangkan algoritma, visualisasi data, dan komputasi numerik. Program MATLAB ini dapat digunakan untuk menyelesaikan masalah komputasi dengan lebih cepat dibandingkan dengan bahasa pemrograman tradisional, seperti C, C++, dan Fortran. MATLAB digunakan untuk banyak aplikasi seperti *signal and image processing*, desain kontrol, pengujian dan pengukuran, *permodelan*, dan analisis.

Simulink merupakan bagian dari MATLAB untuk memodelkan, mensimulasikan, dan menganalisa sistem dinamik. *Simulink* dapat membentuk *model* dari awal atau memodifikasi *model* yang sudah ada sesuai dengan apa yang diinginkan. Selain itu *simulink* juga mendukung sistem *linier* dan *non-linier*, pemodelan waktu kontinyu atau diskrit, atau gabungan. *Simulink* ini dapat digunakan sebagai media untuk menyelesaikan masalah dalam industri nyata meliputi kedirgantaraan dan pertahanan, otomotif, komunikasi, elektronik dan pemrosesan sinyal,

Salah satu modul dalam *Simulink* yang dapat digunakan untuk komunikasi perangkat keras adalah *Instrument Control Toolbox*. Modul ini merupakan kumpulan fungsi *m-file* yang dibangun pada lingkungan komputasi teknis MATLAB. *Toolbox* ini menyediakan kerangka kerja untuk komunikasi instrumen yang mendukung GPIB *interface*, standar VISA, TCP/IP, dan protokol UDP. *Toolbox* ini memperluas fitur dasar *serial port* yang ada dalam MATLAB. Selain itu *toolbox* ini berfungsi untuk komunikasi data antara *workspace* MATLAB dan peralatan lainnya. Data tersebut dapat berbentuk biner atau *text*.

Komunikasi *serial* merupakan protokol dasar tingkat rendah untuk komunikasi antara dua peralatan atau lebih. Pada umumnya satu komputer dengan *modem*, *printer*, mikrokontroler, atau peralatan lainnya. *Serial port* mengirim dan menerima informasi *bytes* dengan hubungan seri. *Bytes* tersebut dikirimkan menggunakan format biner atau karakter ASCII (*American Standard Code for Information Interchange*). Dalam komunikasi serial MATLAB, agar data ASCII dapat diproses *real time*, maka digunakan ASCII *encode* dan *decode* yang terdapat pada *xPC Target Library for RS232*. ASCII *encode* merupakan blok dalam *simulink* yang digunakan untuk mengubah data *bytes* menjadi karakter ASCII. Sedangkan ASCII *decode* merupakan blok *Simulink* yang digunakan untuk mengubah karakter ASCII menjadi data *bytes* yang kemudian dapat dikonversi sesuai kebutuhan.



Gambar 2.3 *Jendela Editor dan Simulink MATLAB*

Halaman ini sengaja dikosongkan

BAB 3 PERANCANGAN SISTEM

Bab ini akan membahas mengenai proses perancangan keseluruhan sistem. Tahapan proses perancangan sistem meliputi perancangan konstruksi *Multistage Axial Field BLDC Motor*, desain *driver* yang menggunakan metode *Self Commutation*, proses identifikasi *Multistage Axial Field BLDC Motor*, perancangan model matematika dan simulasi *plant* berdasarkan hasil identifikasi, perancangan model matematika dan simulasi *PID Adaptive Direct STR (Self Tuning Regulator)*. Gambaran dari sistem secara keseluruhan adalah sebagai berikut.

Proses Identifikasi bertujuan untuk mendapatkan permodelan matematika, agar dapat dilakukan simulasi *BLDC Motor* dengan *Simulink* pada program *MATLAB*.

Simulasi *plant* dibuat dengan acuan parameter tertentu. Parameter konstanta yang diperlukan antara lain adalah resistansi (R), induktansi (L), konstanta torsi (K_{Tm}), konstanta ggl / back e.m.f (K_{ggl}), induktansi bersama atau *mutual inductance* (M), *dampner* (B), dan momen inersia (J) [3].

3.1 Permodelan Motor BLDC

Permodelan matematika *BLDC* didapatkan dari data konstanta parameter yang terdapat pada referensi [3]. Permodelan bertujuan agar *plant dapat* dapat disimulasikan. Model matematika yang digunakan adalah dengan menggunakan Persamaan (2.5).

$$\begin{bmatrix} L & M & L \\ M & L & M \\ M & M & L \end{bmatrix} \begin{bmatrix} \frac{di_u}{dt} \\ \frac{di_v}{dt} \\ \frac{di_w}{dt} \end{bmatrix} = \begin{bmatrix} -R & 0 & 0 \\ 0 & -R & 0 \\ 0 & 0 & -R \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_u \\ V_v \\ V_w \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_u \\ e_v \\ e_w \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} v_n$$

Persamaan di atas dapat kita tulis menjadi,

$$\begin{bmatrix} L & M & L \\ M & L & M \\ M & M & L \end{bmatrix} \begin{bmatrix} \frac{di_u}{dt} \\ \frac{di_v}{dt} \\ \frac{di_w}{dt} \end{bmatrix} = \begin{bmatrix} -R & 0 & 0 \\ 0 & -R & 0 \\ 0 & 0 & -R \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_u - e_U - v_n \\ V_v - e_V - v_n \\ V_w - e_W - v_n \end{bmatrix} \quad (3.1)$$

$$\begin{bmatrix} \frac{di_u}{dt} \\ \frac{di_v}{dt} \\ \frac{di_w}{dt} \end{bmatrix} = \begin{bmatrix} L & M & L \\ M & L & M \\ M & M & L \end{bmatrix}^{-1} \begin{bmatrix} -R & 0 & 0 \\ 0 & -R & 0 \\ 0 & 0 & -R \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + \begin{bmatrix} L & M & L \\ M & L & M \\ M & M & L \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_u - e_U - v_n \\ V_v - e_V - v_n \\ V_w - e_W - v_n \end{bmatrix} \quad (3.2)$$

Dengan Persamaan (3.2) dan parameter yang di dapat pada referensi [9] maka kita dapat melakukan simulasi dengan *Simulink* pada program *MATLAB*.

Konstanta parameter yang digunakan untuk mendesain *plant* tersebut didapat dari referensi penelitian [9] seperti yang dirunjukkan pada Tabel (3.1)

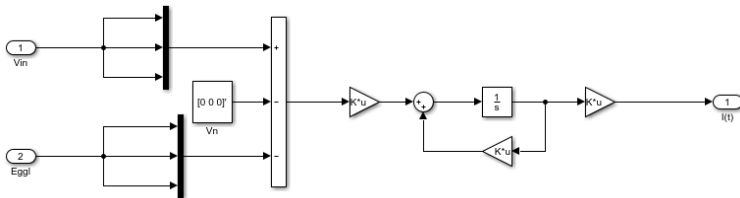
Tabel 3.1 Parameter *Plant*

No.	Nama (Satuan)	Simbol	Nilai
1.	Tahanan stator (ohm)	R	0,87
2.	Induktansi Stator (mH)	L	0,16
3.	Konstanta Torsi (N.m)	K_{Tm}	0,225
4.	Konstanta ggl (Mv.s/r)	K_{ggl}	39,56
5.	Induktansi Bersama μH	M	138,22
6.	Konstanta Damper (N.m.s/R)	B	0,094
7.	Moment Inersia (KG.m ²)	J	0,0102

Motor BLDC terdiri dari bagian elektrik, bagian mekanik, beban (load), dan kontroler.

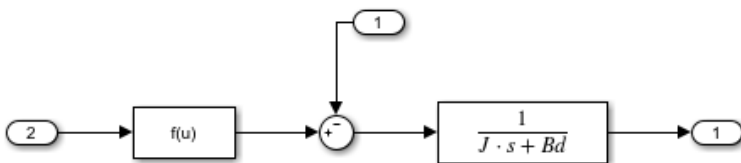
3.1.1. Permodelan Bagian Elektrik dan Mekanik Motor BLDC

Bagian mekanik motor terdiri dari stator dan rotor. Stator adalah bagian yang tidak bergerak, sedangkan rotor adalah bagian yang berputar.



Gambar 3. 1 Diagram Blok *Simulink* Bagian Elektrik Motor

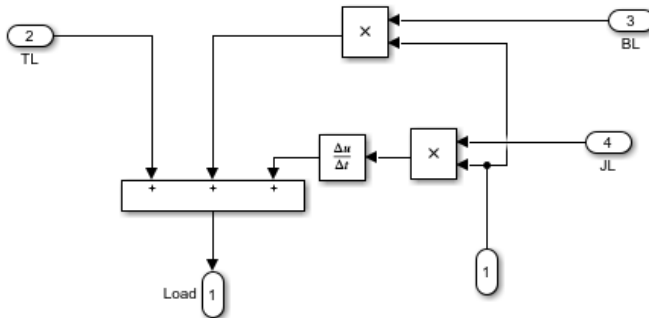
Di dalam stator dan rotor terdapat bagian elektrik, yaitu berisi kumparan stator dan kumparan rotor. Kumparan stator berfungsi menghasilkan medan putar dan kumparan rotor berfungsi untuk membangkitkan medan searah. Ketika motor listrik dihubungkan pada sumber tegangan, pada stator akan terbangkitkan medan berputar dalam periode tertentu. Medan searah pada rotor akan mengikuti medan putar pada stator, sehingga rotor bisa berputar seiring berputarnya medan putar. Diagram blok bagian elektrik motor dapat dilihat pada Gambar 3.1.



Gambar 3. 2 Diagram Blok *Simulink* Bagian Mekanik Motor

3.1.2. Permodelan Beban pada Motor BLDC

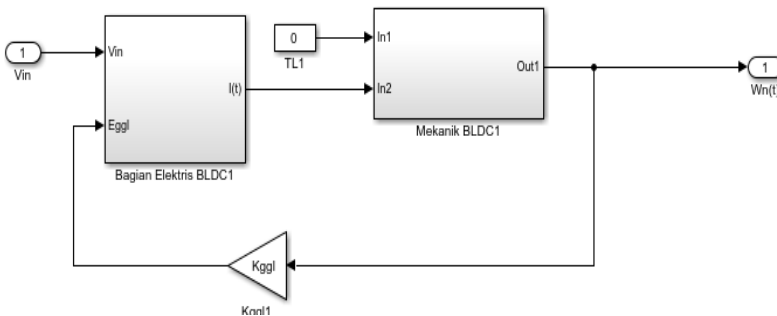
Load atau beban pada motor BLDC terdapat tiga jenis yaitu Torsi *load*, beban *damper* (B), dan beban momen inersia (J). Permodelan beban seperti yang dapat dilihat pada Gambar 3.3



Gambar 3. 3 Diagram Blok *Simulink* Load pada Motor

3.1.3. Permodelan *Nominal plant*

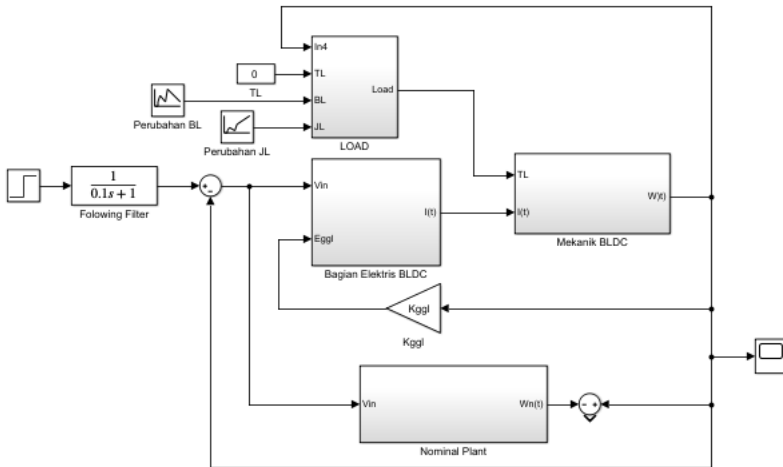
Nominal plant adalah model dinamik *plant* sistem dengan parameter nominal. Dapat berupa model dinamik linier atau non linier sesuai dengan model dinamik *plant*.



Gambar 3. 4 Diagram Blok *Simulink* *Nominal Plant* pada Motor

3.1.4. Diagram Blok *Simulink* Motor BLDC

Setelah perancangan bagian elektrik dan mekanik motor BLDC, maka kita dapat merancang keseluruhan rangkaian motor. Gambar simulasi rancangan motor *BLDC* dapat dilihat pada gambar berikut



Gambar 3.5 Diagram Blok *Simulink* Motor BLDC

3.2 Permodelan Kontroler

STR direct, parameter kontroler diestimasi berdasarkan spesifikasi error model yang akan dipenuhi. Metode ini memerlukan model referensi sebagai acuan untuk mencari nilai error. Pada tugas akhir ini digunakan model referensi orde 1 dengan menggunakan model sebagai berikut:

$$\frac{1}{0,2s + 1} \quad (3.3)$$

Nilai didapatkan dari *trial and error* pada saat merancang kontroler pada *MATLAB*

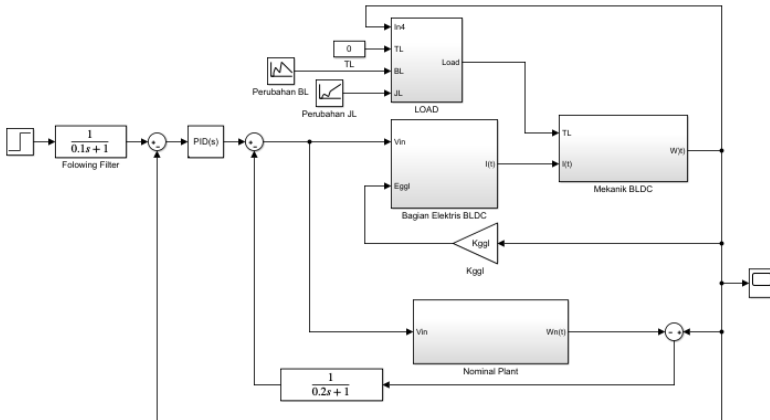
Kontroler dapat berupa kontroler *PID* yang di desain dengan metode tertentu sedemikian rupa sehingga respons sistem *closed loop* memenuhi spesifikasi performansi yang diinginkan. Pada tugas akhir ini dilakukan tuning *PID* dengan menggunakan metode *trial and error* pada *MATLAB*.

3.3 Desain Sistem Pengaturan

Setelah kontroler selesai di desain, maka kita dapat menggabungkan *plant* dengan kontroler sehingga dapat melakukan percobaan simulasi dengan menggunakan kontroler pada *plant*.

Percobaan yang dilakukan menggunakan kontroler pada *plant* adalah menambah beban yang berupa *damper* (B) dan inersia (J) pada *plant* sehingga nanti akan diamati apakah kontroler benar-benar dapat bergufungsi menjaga kecepatan pada *plant*.

Desain gabungan antara *plant* dan kontroler dapat dilihat pada gambar berikut.



Gambar 3. 6 Desain Sistem Pengaturan

BAB 4

SIMULASI SISTEM DAN ANALISA

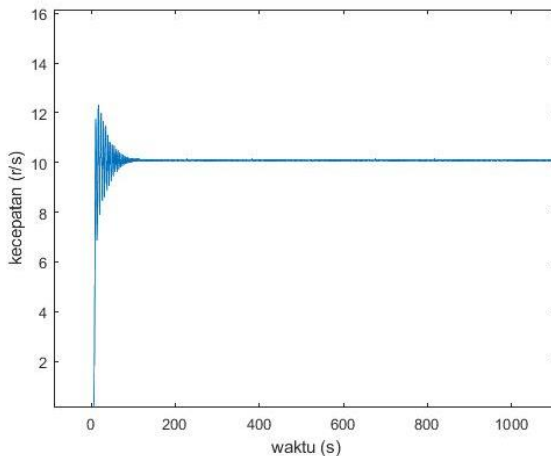
Bab ini berisi tentang hasil simulasi dari perancangan yang sudah dilakukan pada Bab III. Pengujian kontroler juga dengan memberikan gangguan pada sistem. Kemudian dilakukan analisa terhadap data hasil simulasi sistem.

4.1 Simulasi Motor BLDC

Setelah perancangan permodelan *plant*, maka akan didapatkan hasil dengan melakukan pengujian melalui simulasi. Dari hasil pengujian tersebut, dapat diketahui karakteristik dari *plant* tersebut. Proses pengujian dilakukan dengan dua kondisi, yaitu kondisi tanpa kontroler, dan kondisi dengan kontroler.

4.1.1 Kondisi *Open Loop* Tanpa Kontroler dengan Beban Nominal

Percobaan pertama *plant* diberi tegangan 24 VDC dan tidak ditambahkan dengan beban tambahan, hanya beban nominal, yaitu beban yang berasal dari plant itu sendiri. Kontroler juga belum ditambahkan pada plant. Sehingga diperoleh respon kecepatan seperti pada Gambar 4.1.



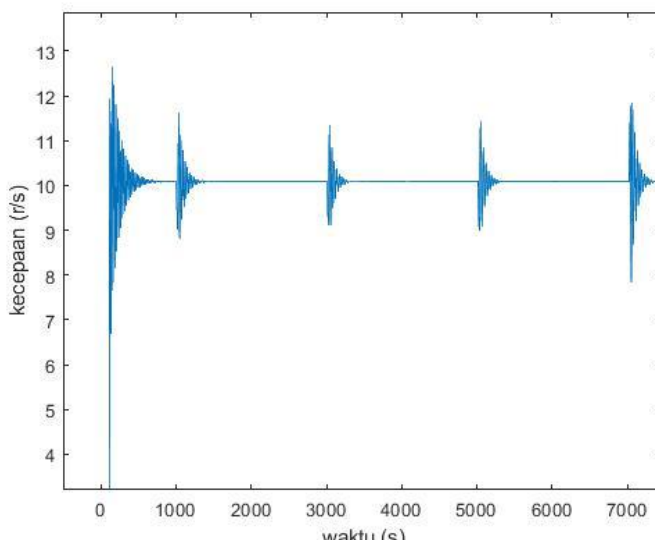
Gambar 4.1 Respon Kecepatan pada Kondisi *Open Loop* dan Tanpa Beban Tambahan

Respon kecepatan pada kondisi *open loop* tanpa kontroler menunjukkan bahwa kecepatan plant tidak dapat mencapai nilai *final value* yang telah ditentukan yaitu kecepatan 50 r/s. Pada percobaan selajutnya dapat kita lakukan uji coba dengan menambahkan beban tambahan untuk melihat respon kecepatan.

4.1.2 Kondisi *Open Loop* Tanpa Kontroler dengan Beban Tambahan

Pada percobaan ini *plant* akan diberi beban tambahan tetapi belum menggunakan kontroler. Respon kecepatan yang diperoleh ditunjukkan pada Gambar 4.2.

Terdapat tiga jenis beban pada *plant* motor *BLDC* yaitu beban torsi, beban *damp*er, dan beban inersia. Pada Percobaan 4.1.2 dan 4.1.4 nilai parameter yang akan diubah yaitu pada beban *damp*er (B) dan beban inersia (J) yang masing masing diberikan tambahan beban sebesar 50% hingga 100%.

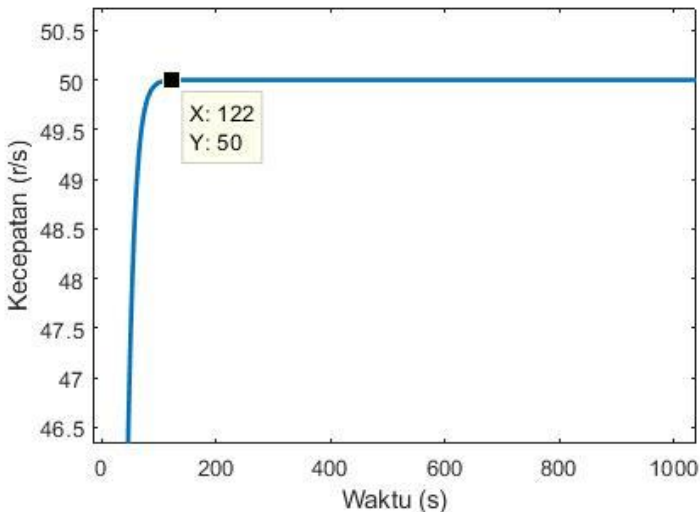


Gambar 4.2 Respon Kecepatan pada Kondisi Open Loop Tanpa Kontroler dan Beban Tambahan

Respon kecepatan yang didapat dari hasil percobaan menunjukkan tidak dapat mencapai *final value* yang telah ditentukan yaitu 50 r/s dan respon kecepatan mengalami osilasi karena adanya penambahan beban. Dari hal ini dapat kita simpulkan bahwa diperlukannya kontroler untuk menjaga respon kecepatan tidak ada gangguan ketika beban bertambah.

4.1.3 Kondisi *Close Loop* dengan Kontroler dengan Beban Nominal

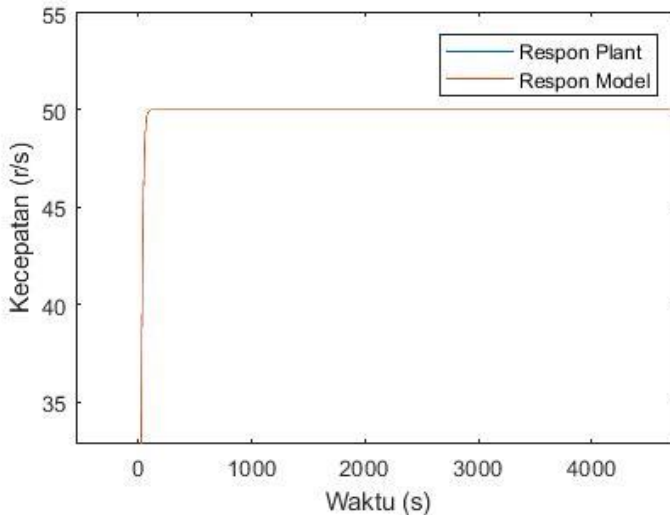
Pada kondisi ini, *plant* ditambahkan dengan kontroler yaitu *Adaptive PID Direct STR*. Pada percobaan ini tidak ditambahkan beban tambahan, melainkan hanya beban nominal, yaitu beban dari *plant* itu sendiri. Respon kecepatan yang diperoleh ditunjukkan pada Gambar 4.3.



Gambar 4. 3 Respon Kecepatan pada Kondisi *Closed Loop* dengan Kontroler dan Tanpa Beban Tambahan.

Berdasarkan Gambar 4.3 dapat kita ketahui bahwa respon kecepatan pada kondisi *closed loop* yaitu 50 r/s sesuai dengan referensi yaitu 50 r/s. selain itu waktu yang dibutuhkan untuk mencapai *steady*

state, yaitu 1,22 detik. Perbandingan respon kecepatan antara *plant* dan *plant* model referensi dapat dilihat pada Gambar 4.4.



Gambar 4. 4 Respon *Output* antara *Plant* Model dan *Plant* Kondisi *Closed Loop* dengan Kontroler

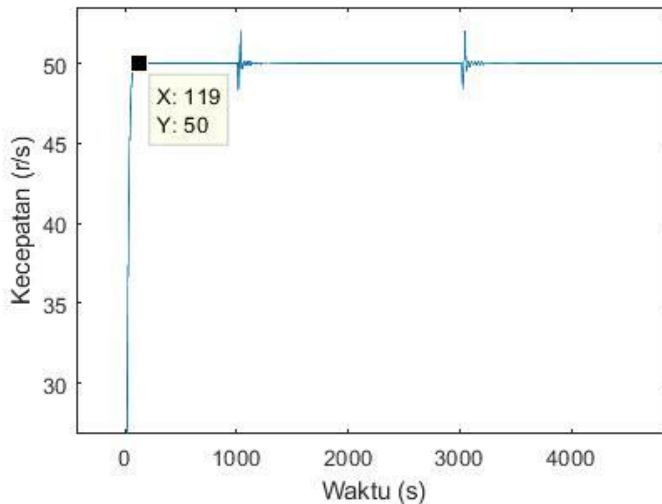
Dari Gambar 4.4, dapat dilihat bahwa respon *output* antara *plant* model referensi dan *plant* mempunyai selisih yang sangat kecil. Dapat disimpulkan bahwa kontroler yang dirancang, mampu menjaga respon sistem tetap stabil.

Pada percobaan selanjutnya dapat kita lakukan percobaan dengan menambah nilai beban tambahan *damper* dan momen inersia

4.1.4 Kondisi *Close Loop* dengan Kontroler dan Beban Tambahan *J load*

Pada simulasi yang dilakukan kali ini, *plant* akan diberi beban sehingga kontroler diharapkan akan berfungsi menjaga kecepatan agar dapat mencapai *final value* yang ditentukan yaitu 50 r/s tanpa adanya gangguan.

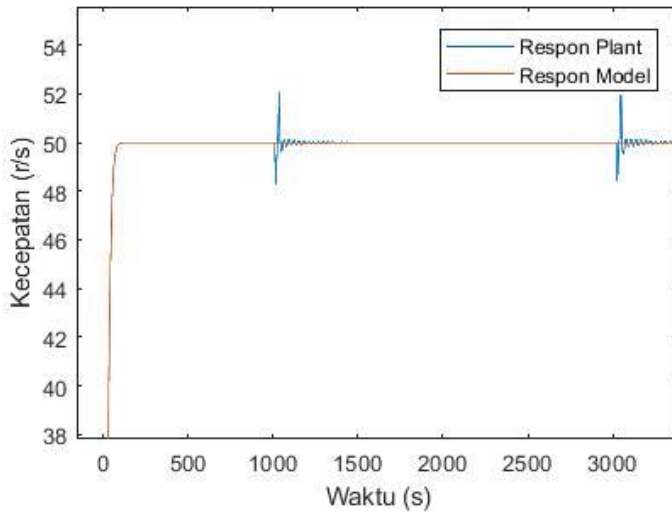
Pada percobaan ini plant juga akan diberi beban tambahan pada damper dan momen inersia untuk menguji kerja kontroler apakah dapat berfungsi dengan baik. Respon kecepatan yang diperoleh ditunjukkan pada Gambar 4.5.



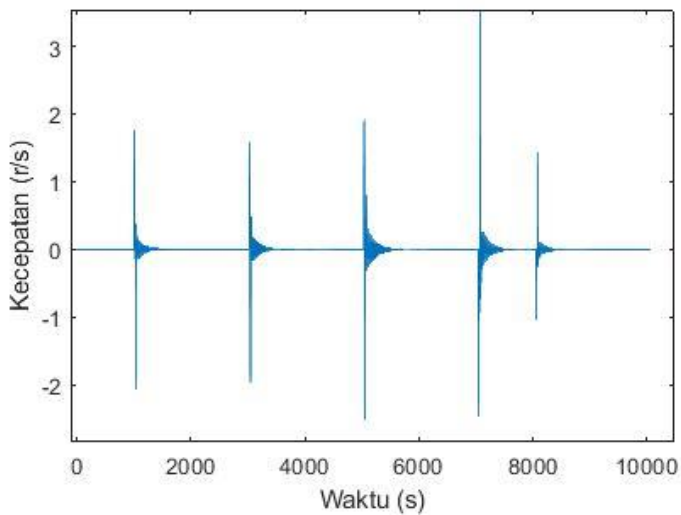
Gambar 4. 5 Respon Kecepatan pada Kondisi *Closed Loop* dengan Kontroler dan Beban Tambahan

Hasil percobaan menunjukkan bahwa respon kecepatan pada kondisi *closed loop* 50 r/s sama dengan referensi yang diinginkan yaitu 50 r/s. Waktu yang dibutuhkan untuk mencapai *steady state*, yaitu 1,19 detik. Respon kecepatan antara *plant* model referensi dan *plant* kondisi *closed loop* dapat dilihat pada Gambar 4.6. Respon kecepatan dari *plant* mendekati dengan respon kecepatan *plant* model referensi.

Masih terdapat gangguan tetapi sangat kecil seperti ditunjukkan pada Gambar 4.7. Dapat kita hitung *Root Mean Squared Error (RMSE)* dari respon kecepatan yaitu 0,0214. Dengan nilai *RMSE* yang mendekati 0 yaitu 0,0214 hal ini membuktikan kontroler berfungsi dengan baik untuk mengatasi gangguan berupa penambahan beban tambahan dari damper dan momen inersia.



Gambar 4. 6 . Respon Kecepatan Antara *Plant* Model Referensi dan *Plant* Kondisi *Closed Loop*



Gambar 4. 7 Selisih Respon *Output* dari *Plant* dan *Plant* Model Referensi

BAB 5

PENUTUP

Pada Bab ini akan dibahas mengenai hasil akhir dari pengerjaan Tugas Akhir meliputi kesimpulan dan saran sebagai referensi tambahan agar penelitian yang akan dilakukan setelahnya dapat lebih baik lagi.

5.1. Kesimpulan

Dari analisa yang telah dilakukan terhadap hasil simulasi maka dapat disimpulkan

1. Perancangan kontroler *direct STR* untuk pengaturan kecepatan pada motor *BLDC* dalam kondisi tanpa tambahan beban (dengan beban nominal) dapat dikatakan berhasil karena menghasilkan respon kecepatan yang dapat mengatasi gangguan dan mencapai kecepatan yang diinginkan yaitu 50 r/s dan mencapai kondisi *steady state* dengan waktu 1,22 detik
2. Pada kondisi *plant* dengan tambahan beban sebanyak 50% hingga 100% pada beban *damper* dan beban inersia. Respon kecepatan memiliki nilai *Root Mean Squared Error (RMSE)* sebesar 0,0214. Dengan nilai RMSE yang relatif kecil atau mendekati 0, maka dapat disimpulkan bahwa perancangan Kontroler berhasil.

5.2. Saran

Karena pada penelitian ini hanya menggunakan kontroler *direct STR* maka untuk penelitian selanjutnya disarankan untuk melakukan perbandingan terhadap kontroler *STR* yang konvensional atau biasa disebut *indirect STR*. Diharapkan pada penelitian selanjutnya didapat perbandingan antara *direct STR* dan *indirect STR*.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Khairi. A, Amien, S, “Analisis Pengaruh Tegangan terhadap Torsi dan Putaran pada Motor Arus Searah Penguat Shunt”, SINGUDA ENSIKOM, Vol. 6 No. 2, February, 2014
- [2] C.-Y. Chen, Member, IEE, W.-C. Chan, T.-C. Ou, S.-H. Yu, and T.-W. Liu, “Sliding Mode Speed Control of Brushless DC Motor Using Pulse-Width- Modulated Current Regulator”, 2009.
- [3] Fakhrlul, Mochamad Rameli, dan Rusdhianto Efendie, “Perancangan Sistem Pengaturan Kecepatan Brushless DC Motor (BLDCM) dengan Self Commutation menggunakan Sliding Mode Control (SMC)” Thesis Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember, 2016.
- [4] Yunita. R, Pramudijanto. J, dan Iskandar. E. “Perancangan dan Implementasi Pengaturan Kecepatan Motor Induksi 3 Fasa dengan PID Self Tuning Regulator (STR)”, Jurnal Teknik Pomits Vol. 2, No. 1, hal 1-7, 2014.
- [5] Amin Mahmoudi, Member, IEEE, Solmaz Kahourzade, Member, IEEE, Nasrudin Abd Rahim, *Senior Member, IEEE*, Wooi Ping Hew, *Member, IEEE*, and Mohammad Nasir Uddin, *Senior Member, IEEE* “ Design, Analysis, and Prototyping of a Novel-Structured Solid-Rotor-Ringed Line-Start Axial-Flux Permanent-Magnet Motor”, 2004.
- [6] I. Boldea, L.N. Tutelea, S.I. Deaconu, F. Marignetti, “Dual rotor single-stator axial air gap PMSM motor/ generator drive for HEVs: a review of comprehensive modeling and performance characterization”, *IEEE*, 2012
- [7] Hongqin Xie, Xubiao Wei, Yang Liu, Yaojing Feng, Yingli Zhang, Xingxing Yang, and Kai Yang, “Research of Asymmetrical Bidirectional Magnet Skewing Technique in Modular Multi-Stage Axial Flux Permanent Magnet Synchronous Motor”, *IEEE*, 2015
- [8] Eduardo Quintero-Manriquez, Ramon A. Felix, “Second-Order Sliding Mode Speed Controller with Anti-windup for BLDC Motors”, *IEEE*, 2014.
- [9] Abe Dharmawan. “Pengendalian Motor Brushless DC Dengan Metode PWM Sinusoidal Menggunakan ATMEGA 16” Skripsi Universitas Indonesia, 2009.

- [10] Takashi Kenjo and Shigenobu Nagamori, "Brushless Motors Advance Theory and Modern Applications" SOGO ELECTRONICS PRESS, 2003.
- [11] Oktafiani, N.A, dkk. "Perancangan Kontroler PID STR (Self Tuning Regulator) Direct untuk Mengendalikan Frekuensi Pembangkit Listrik Mini Hidro. Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember, Institut Teknologi Sepuluh Nopember, 2012.
- [12] Fitriyanto, Muhammad. "Aplikasi Kendali Adaptif pada Pengendalian Plant Pengatur Suhu dengan Self Tuning Regulator (STR)". Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro Semarang. 2014.
- [13] E. F. Camacho, "Model Predictive Control (Advanced Textbooks in Control and Signal Processing)", Springer, Inggris, 2000.
- [14] Fendi Ferdiansyah. "Teori Kontrol PID (Proportional–Integral–Derivative)"[Online]. Tersedia: <http://untirta.academia.edu/FendiFerdiansyah> [Diakses 10 Maret 2018].

RIWAYAT HIDUP



Nitya Ayuning Fasalina adalah nama lengkap penulis. Penulis lahir di kota Probolinggo Jawa Timur pada tanggal 20 Juli 1995 yang merupakan anak pertama dari tiga bersaudara pasangan Joni Anugerahing Adji dan Sri Wahyuningsih. Penulis sedari kecil telah tinggal di kota Malang dan memulai pendidikan dasar di SDN Lesanpuro IV Malang, SMPN 1 Malang dan SMAN 5 Malang. Setelah lulus dari SMA pada tahun 2012, penulis melanjutkan studi di Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya melalui jalur Mandiri pada tahun yang sama. Konsentrasi penulis adalah pada bidang studi Teknik Sistem Pengaturan dan selama kuliah, penulis aktif menjadi pengurus di bidang manajerial pada organisasi mahasiswa jurusan.

e-mail : nityaafasalina@gmail.com